INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. (2010)
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/dac.1165

A scalable and efficient key escrow model for lawful interception of IDBC-based secure communication‡

Kyusuk Han[1,*,†], Chan Yeob Yeun[2], Taeshik Shon[3], Jonghyuk Park[4] and Kwangjo Kim[1]

[1]*Korea Advanced Institute of Science and Technology (KAIST), 119, Munjiro, Yuseong-gu, Daejeon 305-732, South Korea*
[2]*Khalifa University of Science, Technology and Research, P.O. Box 573, Sharjah, United Arab Emirates*
[3]*Convergence S/W Lab, DMC R&D Center, Samsung Electronics, Dong Suwon P.O. Box 105, Maetan-3dong, Suwon, 442-600, Korea*
[4]*Department of Computer Science and Engineering, Seoul National University of Technology, 172, Gongreung 2-dong, Nowon, Seoul, South Korea*

SUMMARY

Key escrowing is one of the core technologies for the lawful interception (LI) of secure communications in the wired and wireless networks. Although many previous studies on the key escrowing have been done before, they are insufficient to be deployed in practical networks due to conflicts with the LI requirements. Moreover, there is lack of consideration on the LI of ID-based cryptosystem (IDBC)-based secure communication because the interest of the LI was moved to the industries and IDBC has the inherent key escrowing property. However, the inherent property of IDBC cannot prevent 'illegal' eavesdropping of all the communications in the networks from the law enforcement agency with the 'legally' obtained key. Thus, we propose a new key escrow model that satisfies the requirements of LI and overcomes the potential threats of IDBC. Our contributions enable the scalable and efficient key escrowing for the LI of secure one-way and two-pass communication in the mobile networks. Copyright © 2010 John Wiley & Sons, Ltd.

Received 28 February 2010; Revised 13 May 2010; Accepted 14 May 2010

KEY WORDS: lawful interception; ID-based cryptosystem; key escrow; mobile network

1. INTRODUCTION

Lawful interception (LI) is inevitably required for protecting the national security or for detecting the criminal evidence, but it should be allowed under strict guidelines and regulations. Several

*Correspondence to: Kyusuk Han, Korea Advanced Institute of Science and Technology (KAIST), 119, Munjiro, Yuseong-gu, Daejeon 305-732, South Korea.
†E-mail: hankyusuk@kaist.ac.kr
‡A part of this paper was presented in IEEE International Conference on Consumer Electronics '09 [1].

technical specifications for the LI such as [2–5] are designed to satisfy such restrictions. For the LI on secure communications, such regulations state that the network service providers should provide the proper decryption method for the request of the law enforcement agency (LEA). Thus, the secret keys of network service subscribers are escrowed and provided for the request of the LEA. After the permission of the LI is expired, it should be disabled that the LEA uses the secret key. Also, the network subscribers should not recognize whether they are under surveillance [3].

While the current security architecture of mobile communication networks widely uses the symmetric cryptosystem that shares secret keys between the subscribers and the service provider [6], the advance of communication technologies has introduced the IP-based communication such as Voice over IP [7, 8], where the communication is not limited to two-pass communications such as the voice and video conversation, but include the one-way data communications such as SMS/MMS and e-mail services.

In such environments, escrowing the session key is not sufficient for supporting the LI of advanced security features such as the digital signature. Thus escrowing the private key is necessary to support the LI of secure one-way communications. For example, using the private key of the receiver can only decrypt the secure e-mail that has been encrypted by using the public key, the private key should be provided to the LEA.

Since the public key has a much longer lifetime than the session key of the symmetric cryptosystem, it cannot be technically prevented from the LEA illegally eavesdropping the communication if the public key has not been updated. Thus, the existing key escrow models focus on limiting the capability of the LEA [9–14]. However, these approaches have the problem that subscribers should participate in escrowing the public key pairs in order to limit the warrant bound of LI using their models. Such processes conflict with the LI requirements such that the subscribers never recognize whether their communications are under surveillance.

Moreover, there is a lack of consideration of the LI using the ID-based cryptosystem (IDBC) [15]. Studies on IDBC were introduced after the interest on the key escrowing model had moved to the industry. Also, the inherent property that the key escrowing is initially available stunted the interest on the key escrowing of IDBC. By using IDBC, the LEA could self-generate the private key of each user from the escrowed master key. However, the inherent property of IDBC for the LI has two significant shortcomings: One is that the LEA can also generate any key without legal permission until the master key is updated, as every subscriber's private key is generated from the master key. The other is that the update of a single private key of a subscriber is infeasible. Thus, the update of the public key pair in IDBC has heavy communication and computational costs.

Therefore, our motivation is to design a new robust and feasible key escrow model for securing communications based on IDBC that not only overcomes the shortcomings of the previous key escrowing models for the LI in the mobile networks, but also enables efficient update of a single private key that reduces the inherent threat of IDBC. Our new model also improves the efficiency in the public key management.

The organization of this paper is as follows: Section 2 briefly shows the network architecture for the LI and the key escrowing models. Section 3 illustrates the existing key escrow model and addresses the shortcomings of the previous key escrow models. Section 4 describes our new scalable and efficient key escrow model. Section 5 analyzes our protocol and compares it with the previous protocols. Finally, we conclude our paper in Section 6.

DOI: 10.1002/dac

2. LI OF SECURE COMMUNICATION

In this section, we briefly describe the current standard LI architectures in mobile networks, and the key escrowing models for the LI of secure communications. We will use some notations as in Table I in the paper.

2.1. Mobile network architecture for LI

While the generic LI architectures are largely specified by ANSI, ETSI, 3GPP, and etc., we briefly introduce the specification by 3GPP due to the similarity of the architectures. 3GPP specifies the requirements of the LI [3], the architectures and the functions [4], and the handover interface (HI) between the LEA and the MO [5].

Figure 1 shows several HIs that link the LEMF of the LEA to the IIF of the MO. We can assume the HI as the secure channel. Each HI is defined to send the following information: HI1, HI2, and HI3 send administrative information, IRI and CC, respectively.

Table I. Notations.

LI	Lawful Interception	LEA	Law Enforcement Agency
IDBC	ID-based Cryptosystem	CC	Content of Communication
MO	Mobile service Operator	KGC	Key Generation Center
HI	Handover Interface	HI1	Handover Interface 1
HI2	Handover Interface 2	HI3	Handover Interface 3
LEMF	Law Enforcement Management Function	EA	Key Escrow Agency
IIF	Internal Network Interception Function	IRI	Intercept Related Information

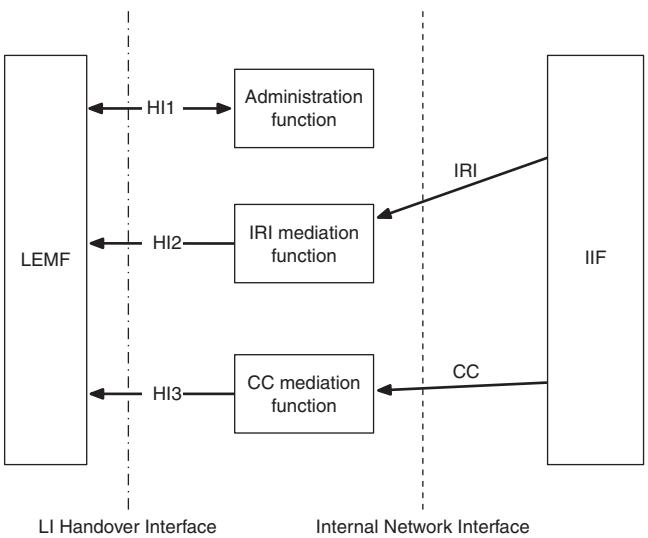

Figure 1. Architecture for lawful interception by 3GPP.

DOI: 10.1002/dac

The administrative function in HI1 includes the network management function. Both IRI and CC are sent to LEMF via the IIF of the MO. The LEA manages the LEMF that gathers and analyzes the information of both IRI and CC. IRI is coded using ASN.1 and transmitted from IIF of the MO to LEMF via HI2.

When the LEA requests the LI of the secure communication to the MO via HI1, the MO may provide the proper decryption method (the escrowed keys) via HI2 and the encrypted communication via HI3.

2.2. Key escrowing for the LI of the secure communication

In the current symmetric cryptosystem-based security architecture [4], the MO also plays the role of the EA that provides the session keys with short lifetimes for the LEA. Thus, most studies on the key escrowing are for the public key infrastructure (PKI)-based secure one-way communications such as secure e-mail.

The key escrow model for the PKI-based secure communication consists of the EA and the LEA: the EA stores users' private keys, the LEA requests the private keys for the purpose of the LI under the legal permission of the court. LI procedures are described in brief as follows: When the users initiate a secure communication, denoted by $\mathscr{C}$, the LEA are granted LI of $\mathscr{C}$. Then the LEA requests the escrowed key to the EA, and the EA provides the key to the LEA. Finally, the LEA discloses the information of $\mathscr{C}$.

As there is potential vulnerability of the malicious behavior of the EA or the LEA, most studies have focused on limiting the capability of the EA and the LEA. Micali [9] proposed the protocol that a user divides his private key into several pieces and registers it to several EAs in order to limit the capability of the single EA. Therefore, the initial key can only be recovered when all EAs agree on the key recovery. Shamir [10] proposed the partial key escrow method that requires sufficient time consumption to protect the incident misuse from the malicious EA. However, this method requests the large overhead that conflicts with the LI requirements [3]. Also, Jefferies *et al.* [11] proposed the warrant bound to limit the duration of the LI of the LEA in order to prevent the malicious behavior of the LEA. For this purpose, Verheul *et al.* [12] proposed fraud detectability while Frankel *et al.* [13] introduced compliance certification.

2.2.1. Abe and Kanda's key escrow protocol. In 2002, Abe and Kanda [14] defined the requirements for the key escrowing and proposed the PKI-based key escrow algorithm for the one-way communication that allows the limited permission period. Their protocol consists of the registration phase, the communication phase, and the disclosure phase as follows:

Registration: A user u generates public key pairs (x_{u_i}, y_{u_i}) for $i = 0, 1, \ldots, t$ and sends to the EA. $x_{u_i} \in_R \mathscr{Z}_q$ is the private key randomly chosen by u and $y_{u_i} := g^{x_{u_i}}$ is the corresponding public key, where g is the generator of $\mathscr{G}_q$, a multiplicative subgroup of order q in Z_p. After verifying the keys, the EA stores everything received.

Communication: u initiates the secure communication $\mathscr{C}_{u_\tau}$ using x_{u_τ} or y_{u_τ}, where τ is the target term wherein monitoring is approved.

Disclosure: the LEA is granted LI of u and discloses $\mathscr{C}_{u_\tau}$ within the warrant (the user u and the term τ).

However, each subscriber has to participate in the key escrowing that conflict with the LI requirements that the subscriber shall not recognize whether they are under surveilance.

DOI: 10.1002/dac

2.3. Initial key escrowing property under IDBC

Although the concept of IDBC was first proposed by Shamir [15] in 1984, the practical IDBC-based models began to be widely studied after Boneh and Franklin [16] proposed the encryption schemes in 2002.

IDBC can be computed using the following properties of pairing: Let the additional group be G_1 and the multiplicative group be G_2. We know that solving the discrete logarithm problem in these groups is hard. Let P be the generator in the additional group. And, let $\hat{e}: G_1 \times G_1 \rightarrow G_2$ be the bilinear pairing satisfying the following properties.
Bilinearity: For all $P, Q \in G_1$ and all $a, b \in Z$, $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$.
Non-degeneracy: For all $Q \in G_1$, if $\hat{e}(P, Q) = 1$, then $P = O$.
Computability: For all $P, Q \in G_1$, there is an efficient algorithm to compute $\hat{e}(P, Q)$.

The generic PKI models require the certificate management that checks the validity of the public key since the public key pair is generated by a prime number. On the contrary, IDBC does not require the certificate management due to the public key being generated from the identity of the user.

2.3.1. Initial key escrow property of IDBC. The public key distribution in IDBC works as follows: For a user A, the KGC generates a master key s, where $s \in Z_P^*$, and computes A's public key $H(\mathrm{ID}_A)$ and A's private key $sH(\mathrm{ID}_A)$ using A's unique ID_A and a hash function $H: Z_P^* \rightarrow G$. Extract s from $H(\mathrm{ID}_A)$ and $sH(\mathrm{ID}_A)$ has the same computational complexity as solving Elliptic Curve Discrete Logarithm Problem (ECDLP) [17] as follows:
Elliptic Curve Discrete Logarithm: With given $P, P' \in G_1$, compute the integer n satisfying $P = nP'$.

Using the master key s securely stored in the KGC enables the generation of all user's private keys. In general, KGC can also be the escrow agency. Thus, IDBC has the initial key escrowing property by storing the master key in the KGC without any additional key escrowing process.

3. SHORTCOMINGS ON THE PREVIOUS KEY ESCROW MODELS

3.1. Conflicting with requirements of LI

In some previous key escrow models [14], all subscribers could self-generate their public key pairs and register to the EA. However, the subscriber's participation in the key escrow procedures fundamentally conflicts with the requirements of the LI that the subscribers never recognize whether they are under surveillance.

3.2. Warrant bound of LEA

In order to provide the proper decryption method, the mobile service providers escrow subscriber's key to the EA and send the escrowed keys for the request of the LEA. Providing the symmetric session key for the secure two-pass communication such as voice conversation has less complication due to the short lifetime of the key that expires after the session is closed.

On the other hand, the private key should be sent to the LEA for the LI of the one-way communication such as secure e-mail. Owing to the lifetime of the public key pair (the public key and the private key) being much longer than that of the symmetric session key, the LEA might

DOI: 10.1002/dac

be able to illegally eavesdrop the subscriber's communication after the permission was expired, if the public key pair was not updated. For example, the permission terms on the LI may be at least several days while the lifetime of public key is about a year in general. Even though a few models such as [14] overcome such a problem, they require the participation of the subscriber that conflicts with the requirements of the LI. Moreover, Abe and Kanda's model [14] only supports the LI of one-way communication.

3.3. Overhead for the network

In existing mobile networks such as 3GPP, the security architecture based on the symmetric cryptosystem is widely adopted in practice due to their performance efficiency. Thus, the previous public key escrow models such as [9, 10, 12, 14] require large overheads on the public key management and the key storage from the non-standard architecture for each key escrow model. Thus, complex network facilities that increase the overall cost of the networks are required.

Applying IDBC, such overheads from the public key management are not required, as the LEA can self-generate the private key of each subscriber with the escrowed master key.

3.4. Security threats on the initial property of IDBC

However, only depending on the inherent property of IDBC has the potential security threat that the LEA can illegally eavesdrop all the communication in the network. When the EA provides the master key s to the LEA for the LI of A, the LEA can generate the private key of A, $sH(\mathrm{ID}_A)$ using the publicly known hash function $H : Z_P^* \rightarrow G$ and A's ID, ID_A. However, the LEA can also compute $sH(\mathrm{ID}_D)$ to eavesdrop another user D without legal permission if the domain master key s was not updated. Moreover, the key update of a single subscriber is not available in IDBC. Once the private key of a subscriber is compromised or known to the LEA, all keys of all subscribers must be updated.

Although several studies such as [18] prevent key escrowing, they cannot be used for the LI from the requirements [3] since they drop the key escrow property of IDBC.

4. PROPOSED KEY ESCROW MODEL

In this section we propose our key escrow model for IDBC-based secure communication that overcome the shortcomings shown in Section 3. We define the following entities in our model:

$\mathscr{L}$: LEA requests the content of the communication and receives the intercept-related information and content from MO under the law.

$\mathscr{M}$: MO requests offers from the mobile communication service including the encryption to subscribers, and provides the proper decryption method and interception-related information for the request of $\mathscr{L}$.

$\mathscr{K}$: KGC requests and provides keys for encryption to MO, subscriber, and $\mathscr{L}$. It also provides the subscribers key for the request of $\mathscr{L}$. (Note that KGC is also the escrow agency.)

Subscriber: Subscriber uses the mobile communication service, and receives the encryption key from KGC.

We assume that the LEA may illegally intercept the secure communication of the subscriber over the warrant bound.

DOI: 10.1002/dac

4.1. Security requirements

The requirements of key escrowing are defined in [14]. Designing the new key escrow model based on the IDBC, we define additional requirements as follows:

- *Non-Subscriber participation*: It shall not be recognized by a subscriber in escrowing and providing subscriber's key to $\mathscr{L}$.
- *Warrant Bounds*: It shall be available to limit the duration of the permission for the LI by $\mathscr{L}$.
- *Key Escrow Efficiency*: It shall not consume large overhead for providing the key to $\mathscr{L}$.
- *Off-line KGC*: When $\mathscr{L}$ obtains the private key or the necessary information for decryption, it should be able to intercept the communication without the help of $\mathscr{M}$.

4.2. Overall key escrow protocol

In this section, we show the key escrow model that enables $\mathscr{L}$ to intercept all kinds of secure communications between two subscribers A and B. Sections 4.2.1 and 4.2.2 show the LI for the two-pass communication and the one-way communication, respectively. The symbol '*I*' denotes the interception procedures while the symbol '*S*' denotes the communication procedures between subscribers.

For the pre-procedures, $\mathscr{M}$ initially operates the key distribution process as in Section 2.3. Thus, we assume that A already stored $sH(\mathrm{ID}_A)$ as the private key and the $H(\mathrm{ID}_A)$ as the public key, while B stores $sH(\mathrm{ID}_B)$ and $H(\mathrm{ID}_B)$. The shared key k_A between $\mathscr{M}$ and A, and k_B between $\mathscr{M}$ and B are initially shared.

We let that $\mathscr{L}$ requests the LI of A to $\mathscr{M}$.

4.2.1. LI for two-pass communication. Let A initiates the secure communication with B and $\mathscr{L}$ are on the surveillance of A.

I.1. $\mathscr{L}$ requests $\mathscr{K}$ and $\mathscr{M}$ for the LI of B via HI1.

S.1. A generates the random integer r_A and the corresponding signature $\mathrm{sign}_A(r_A)$. A encrypts them with the shared key with $\mathscr{M}$, k_A and sends u_1 to the $\mathscr{M}$.

$$u_1 = e_{k_A}(A||B||r_A||\mathrm{sign}_A(r_A))$$

The symbol of $e_k(msg)$ denotes the encryption function and sign is a signature function. Suffixes of each function denote the owner of the key used for the encryption or signing. For example, e_{k_A} denotes the encryption with the shared key between A and $\mathscr{M}$ and sign_A denotes the signature using A's private key. $\|$ denotes concatenation.

S.2. After decrypting u_1, $\mathscr{M}$ verifies r_A with the signature $\mathrm{sign}_A(r_A)$. And then $\mathscr{M}$ encrypts them using k_B and sends u_2 to B.

$$u_2 = e_{k_B}(A||B||r_A||\mathrm{sign}_A(r_A))$$

If $\mathscr{M}$ includes the signature, then $\mathscr{M}$ sends u_2^* to B.

$$u_2^* = e_{k_B}(A\|B\|r_A\|\mathrm{sign}_A(r_A)\|\mathrm{sign}_{\mathscr{M}}(r_A\|\mathrm{sign}_A(r_A)))$$

S.3. After decrypting u_2, B verifies r_A with $\mathrm{sign}_A(r_A)$, and selects another random nonce r_B. Then B generates the signature of r_B, $\mathrm{sign}_B(r_B)$, and sends u_3 to the $\mathscr{M}$, and then, B computes

DOI: 10.1002/dac

$v = \mathrm{dev}f(r_A, r_B)$, where $\mathrm{dev}f$ is a derivation function from r_A and r_B, e.g. the general computation like $+$ or $\times$.

$$u_3 = e_{k_B}(B \| A \| r_B \| \mathrm{sign}_B(r_B))$$

S.4. $\mathscr{M}$ decrypts u_3 and verifies r_B with $\mathrm{sign}_B(r_B)$. Then $\mathscr{M}$ generates u_4 and sends it to A.

$$u_4 = e_{k_A}(B \| A \| r_B \| \mathrm{sign}_B(r_B))$$

S.5. A computes $v = \mathrm{dev}f(r_A, r_B)$. $\mathscr{M}$ also computes v.
I.2. $\mathscr{M}$ sends v with A's ID and the request of the LI to $\mathscr{K}$.
I.3. $\mathscr{K}$ sends $v \cdot sH(\mathrm{ID}_A)$ to $\mathscr{L}$ via HI2. $v \cdot sH(\mathrm{ID}_A)$ denotes the multiplication of v and $sH(\mathrm{ID}_A)$.
I.4. $\mathscr{M}$ sends the IRI via HI2 and the CC to the LEA via HI3, as in Section 2.1.

We use a key agreement protocol between A and B as follows: A computes $k_{AB} = e(v \cdot sH(\mathrm{ID}_A), H(\mathrm{ID}_B))$, whereas B computes $k_{BA} = e(H(\mathrm{ID}_A), v \cdot sH(\mathrm{ID}_B))$. The correctness of the two equations can be shown from the following equation:

$$k_{AB} = e(v \cdot sH(\mathrm{ID}_A), H(\mathrm{ID}_B)) = e(H(\mathrm{ID}_A), H(\mathrm{ID}_B))^{v \cdot s} = e(H(\mathrm{ID}_A), v \cdot sH(\mathrm{ID}_B)) = k_{BA}$$

$\mathscr{L}$ can compute $H(\mathrm{ID}_A)$ and $H(\mathrm{ID}_B)$ with public hash function $H : Z \to P$, and each subscribers identity ID_A and ID_B. Also with $vsH(\mathrm{ID}_A)$, $\mathscr{L}$ can compute k_{AB} for decrypting the secure communication between A and B. k_{AB} is used as the session key between A and B.

Figure 2 depicts the overall process of the LI for two-pass communication.

4.2.2. LI for one-way communication. In this section, we show the model for one-way communication such as e-mail. Let A generates an e-mail message M_A and securely sends it to B. As most steps are the same as the case for two-pass communication, we only describe the differences.

S.1′. A generates the random integer r_A and the corresponding signature $\mathrm{sign}_A(r_A)$. A also encrypts the message M_A with the temporary public key of B, $r_A H(\mathrm{ID}_B)$, which is denoted as $\mathrm{Enc}_B(M_A)$. In this case, only r_A is used due to the one-way communication from A to B. After that A encrypts them with the shared key with MO, k_A and sends u_1 to the MO.

$$u_1 = e_{k_A}(A \| B \| \mathrm{Enc}_B(M_A) \| r_A \| \mathrm{sign}_A(r_A))$$

S.2′. After decrypting u_1, $\mathscr{M}$ verifies r_A with the signature $\mathrm{sign}_A(r_A)$. Then $\mathscr{M}$ encrypts them and sends u_2 to B.

$$u_2 = e_{k_B}(A \| B \| \mathrm{Enc}_B(M_A) \| r_A \| \mathrm{sign}_A(r_A))$$

S.3′. B decrypts u_2 and verifies r_A with $\mathrm{sign}_A(r_A)$. After that B generates $r_A \cdot sH(\mathrm{ID}_B)$ and decrypts $\mathrm{Enc}_B(M_A)$.
I.2′. $\mathscr{M}$ sends r_A with B's ID and the request of LI to $\mathscr{K}$.
I.3′. $\mathscr{K}$ sends $r_A \cdot sH(\mathrm{ID}_B)$ to $\mathscr{L}$ via HI2.
I.4′. $\mathscr{M}$ sends the IRI and the CC to $\mathscr{L}$ via HI2 and HI3.

$\mathscr{L}$ with $r_A \cdot sH(\mathrm{ID}_B)$ cannot extract s even though $\mathscr{L}$ obtains $sH(\mathrm{ID}_A)$ or $sH(\mathrm{ID}_B)$ with r_A from the subscriber A or B from the computational hardness of ECDLP [17]. In case that A receives the secure e-mail from any entities, $\mathscr{K}$ sends $r \cdot sH(\mathrm{ID}_A)$ to $\mathscr{L}$ in I.3', where r is a random nonce.

DOI: 10.1002/dac

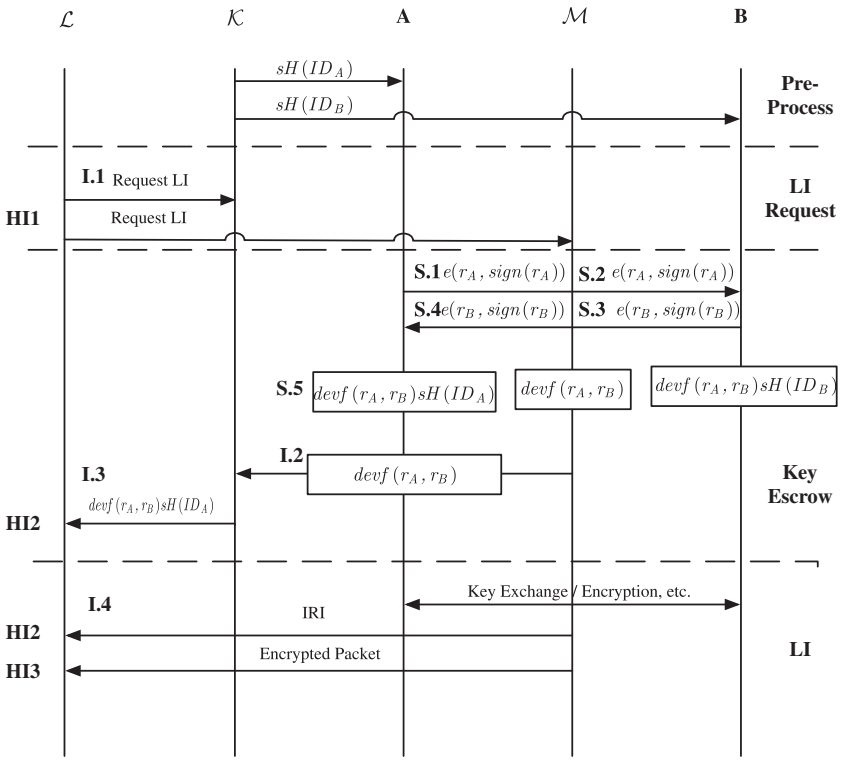

Figure 2. LI procedures for two-pass communication.

5. ANALYSIS OF PROPOSED PROTOCOL

5.1. Security analysis

In this section, we briefly show that our model satisfies the requirements of the LI as follows:

Non Subscriber Participation: In our protocol, since subscribers do not participate in the key escrowing, they cannot recognize whether their communication is under surveillance.

Warrant bounds: The nonces r_A and r_B are randomly selected in each session to prevent the replay attack due to checking the freshness of a session. The private key of the subscriber provided to $\mathscr{L}$ is also different in each session. Consequently, $\mathscr{L}$ fails to eavesdrop the communications in an unauthorized session.

Key Escrow Efficiency: While the generic PKI-based key escrow models require that $\mathscr{K}$ stores the large number of public key pairs, in our protocol $\mathscr{K}$ only stores one master key.

Off-line KGC: After the key escrowing, $\mathscr{L}$ could directly intercept the secure communication via the mobile network operator [5], and reveal the information under surveillance without $\mathscr{K}$.

Our protocol also guarantees key escrow requirements in [14] such as the 'admissibility' that $\mathscr{L}$ verify the message from the subscriber, the 'fraud detectability' that $\mathscr{L}$ can verify the signature of

DOI: 10.1002/dac

random r_A and r_B for checking the freshness, and the 'sender authentication' that $\mathscr{L}$ authenticate the sender from the public key of $H(\mathrm{ID})$.

We also show the security of our protocol when $\mathscr{L}$ illegally eavesdrops the secure communication. If $\mathscr{L}$ tries to intercept the communication without permission, then $\mathscr{L}$ will not receive any support from $\mathscr{M}$ and $\mathscr{K}$. We could consider the following attack scenarios: The LEA attempts the unauthorized interception (eavesdropping) without the legal permission. $\mathscr{L}$ attempts the interception using v after the permission is expired. $\mathscr{L}$ colludes with users A or B to retrieve the master key.

5.1.1. Case 1: *$\mathscr{L}$ attempts the unauthorized interception without any legal permission.* Assume that $\mathscr{L}$ intercepts the encrypted communication between users A and B. The eavesdropping processes are as follows:

1. The user A sends the random number r_A and the signature to the server.
2. The server verifies r_A and the signature of A, and sends r_A and the signature to B.
3. After verifying r_A, B generates the random number r_B and the signature of B, and sends them to the server.
4. The server verifies r_B and the signature of B, and sends them to A.
5. $\mathscr{L}$ attempts to eavesdrop the secure communication.

Let the key agreement protocol be between A and B. A computes $e(v \cdot sH(\mathrm{ID}_A), H(\mathrm{ID}_B))$ whereas B computes $e(H(\mathrm{ID}_A), v \cdot sH(\mathrm{ID}_B))$. In this case, $\mathscr{L}$ has no information of the secret parameter s that is necessary to compute $v \cdot sH(\mathrm{ID}_A)$ and $v \cdot sH(\mathrm{ID}_B)$. Thus, $\mathscr{L}$ cannot know any information of the session key between A and B and cannot decrypt the encrypted packet from the illegal eavesdropping. $\mathscr{L}$ also fails on the attack without knowing $r_A \cdot s$ for the one-way communication.

5.1.2. Case 2: *$\mathscr{L}$ attempts the interception using v after the permission was expired.* Assume that $\mathscr{L}$ tries the unauthorized interception with expired $v \cdot sH(\mathrm{ID}_A)$ as follows: A and B begin another secure communication with a new session.

1. User A sends the random number n_A and the signature to the server.
2. The server verifies n_A and the signature from A and sends them to B.
3. After verifying n_A, B generates the random number n_B and the signature, and sends them to the server.
4. The server verifies n_B and the signature, and sends them to A.
5. $\mathscr{L}$ attempts to eavesdrop the communication with the expired $v \cdot sH(\mathrm{ID}_A)$.

The key agreement protocol between A and B works as follows: After both A and B compute $v' = \mathrm{dev}f(n_A, n_B)$, A computes $e(v' \cdot sH(\mathrm{ID}_A), H(\mathrm{ID}_B))$ whereas B computes $e(H(\mathrm{ID}_A), v' \cdot sH(\mathrm{ID}_B))$. In this case, $\mathscr{L}$ cannot know $v' \cdot sH(\mathrm{ID}_A)$ from $v \cdot sH(\mathrm{ID}_a)$. Thus, $\mathscr{L}$ has no information of the session between A and B, and cannot decrypt the encrypted packet from the packet sniffing. The security of $v' \cdot sH(\mathrm{ID}_A)$ is based on the computational infeasibility of ECDLP [17].

5.1.3. Case 3: *$\mathscr{L}$ colludes with the user A or the user B.* Assume $\mathscr{L}$ has $v \cdot sH(\mathrm{ID}_A)$ and get v and $sH(\mathrm{ID}_A)$ from the colluded user. $\mathscr{L}$ may try to retrieve s from $v \cdot sH(\mathrm{ID}_A)$. However, knowing s from $v \cdot sH(\mathrm{ID}_A)$, v, and $sH(\mathrm{ID}_A)$ has the same computational infeasibility of ECDLP.

DOI: 10.1002/dac

Table II. Comparison of key escrow models.

	Symm. cryptosystem model	Abe-Kanda [14]	Proposed
Warrant bounds	X	O	O
One-way Comm.	X	O	O
Two-pass Comm.	O	X	O
Non Subscriber Participation	O	X	O
Efficiency (Number of Keys)	1	$t+1$	1
Scalability	X	X	O

5.2. Comparisons

In this section, we compare our proposed protocol with the symmetric cryptosystem-based model and Abe–Kanda's protocol [14]. The symmetric cryptosystem-based model only partially satisfies 'warrant bound' with a short lifetime of the key. Abe–Kanda's model does not satisfy 'Non subscriber participation' due to the subscriber self-generating n number of partial public keys and registering them to the escrow agencies. Our protocol is more efficient than the previous models because our model requires only one key, optionally one additional symmetric key, whereas Abe–Kanda's protocol [14] requires $t+1$ number of secret keys, where t is the threshold. Moreover, our protocol provides the LI of both two-pass communication and one-way communication, whereas the symmetric cryptosystem-based model only provides the LI of two-pass communication, and [14] only provides the LI of the one-way communication. Finally, our protocol can be widely used in combination with other key agreement protocols such as Diffie–Hellman protocols, whereas Abe–Kanda's protocol [14] can only be used with their own protocol.

Table II shows the comparison of our proposed protocols with the symmetric cryptosystem-based model and Abe–Kanda's model.

6. CONCLUSION

While the LI is inevitable for protecting the national security or for detecting the criminal evidence, the role of the LEA should be strictly limited in order not to infringe one's rights. Key escrowing is required to surveil the secure communication, and should follow the requirements of the LI.

However, the previous key escrowing protocols are not sufficient to be deployed to the practical LI as they require large overhead and the participation of subscribers that conflict with the requirements in [3]. Although IDBC enables the efficient LI due to their initial key escrow property, it has the potential threat that the LEA is technically able to eavesdrop the secure communication illegally.

In this paper, we proposed the secure and robust key escrow protocol that enables the LI for the secure communication based on IDBC. Providing the warrant bound, our protocol overcomes the security threats from the inherent key escrow property of IDBC and also satisfies the requirements of the requirements in [3], whereas the most previous key escrow models cannot meet the requirements. From the comparison of key escrow models, we found that the proposed protocol provides the scalable and efficient key escrowing for both two-pass and one-way secure communications in mobile networks.

DOI: 10.1002/dac

REFERENCES

1. Han K, Yeun CY, Kim K. New key escrow model for the lawful interception in 3GPP. *2009 IEEE ICCE 2009*, Las Vegas, U.S.A., 12–14 January 2009.
2. Baker F, Foster B, Sharp C. Cisco architecture for lawful intercept in IP networks. *RFC 3924*, 2004.
3. 3rd Generation Partnership Project. *TS 33.106*: *'Lawful Interception Requirements'. V8.1.0 edn*, March 2008.
4. 3rd Generation Partnership Project. *TS 33.107*: *'Lawful Interception Architecture and Functions'. V8.8.0 edn*, June 2009.
5. 3rd Generation Partnership Project. *TS 33.108*: *'Handover Interface for Lawful Interception* (*LI*)*'. V8.7.0 edn*, June 2009.
6. 3rd Generation Partnership Project. *TS 33.102 v9.1.0 3G Security*: *Security Architecture* (*Release 9*), 18 December 2009.
7. Karpagavinayagam B, State R, Festor O. Monitoring architecture for lawful interception in VoIP networks. *Second International Conference on Internet Monitoring and Protection* (*ICIMP 2007*), San Jose, CA, 1–5 July 2007. ISBN 0-7695-2911-9.
8. Seedorf J. Lawful interception in P2P-based VoIP systems. *Principles*, *Systems and Applications of IP Telecommunications. Services and Security for Next Generation Networks*: *Second International Conference*, *IPTComm 2008*, Heidelberg, Germany, July 2008; 217–235. Revised Selected Papers.
9. Micali S. Fair public key cryptosystems. *Proceedings of the Advances in Cryptology-CRYPTO '92*, Santa Babara, CA. Lecture Notes in Computer Science, vol. 740. Springer: Berlin, 1992; 113–138.
10. Shamir A. Partial key escrow: a new approach to software key escrow. *Proceedings of the Key Escrow Conference*, Washington DC, 15 September 1995.
11. Jefferies N, Mitchell C, Walker M. A proposed architecture for trusted third party services. *Proceedings Cryptography*: *Policy and Algorithms*, Brisbane, Australia. Lecture Notes in Computer Science, vol. 1029. Springer: Berlin, 1995; 98–104.
12. Verheul R, van Tilborg, Henk CA. Binding ElGamal: a fraud-detectable alternative to key escrow proposals. *Proceedings of the Advances in Cryptology—EUROCRYPT '97*. Lecture Notes in Computer Science, vol. 1233. Springer: Berlin, 1997; 119–133.
13. Frankel Y, Yungm M. Escrow encryption systems visited: attacks, analysis and designs. *Proceedings of the Advances in Cryptology-CRYPTO '95*. Lecture Notes in Computer Science, vol. 963. Springer: Berlin, 1995; 222–235.
14. Abe M, Kanda M. A key escrow scheme with time-limited monitoring for one-way communication. *British Computer Society 2002*, *The Computer Journal* 2002; **45**(6):661–671.
15. Shamir A. Identity-based cryptosystems and signature schemes. *Proceedings of the Advances in Cryptology—CRYPTO 84*. Lecture Notes in Computer Science, vol. 196. Springer: Berlin, 1984; 47–53.
16. Boneh D, Franklin M. Identity based encryption from the weil pairing. *SIAM Computing* 2003; **32**(3):586–615.
17. Blake I, Seroussi G, Smart N. *Elliptic Curves in Cryptography*. London Mathematical Society, vol. 265. Cambridge University Press: Cambridge, 1999.
18. Hou M, Xu Q, Guo S. Secure key-escrowless explicit authenticated key agreement protocol in ID-based public key cryptography. *Journal of Information Assurance and Security* 2010; **5**:130–137.

AUTHORS' BIOGRAPHIES

Kyusuk Han received the BS degree in Mechanical Engineering from Hongik University, Korea and the MS degree in Computer Science from Information and Communications University, Korea, respectively, in 2001 and 2004. He will receive the PhD in Information and Communications Engineering, KAIST, Korea in Aug. 2010. His interests are in mobile network and sensor network security.

DOI: 10.1002/dac

Chan Yeob Yeun received an MSc and a PhD in Information Security from Royal Holloway, University of London, respectively. After his PhD, he joined Toshiba TRL in Bristol, UK as a Senior Researcher and also worked for LG Electronics, Mobile Communications Research Lab in Seoul, Korea as a Research Fellow. Then, he joined Information and Communications University (ICU) as a Research Professor until August 2008. Currently, he enjoys researching Ubiquitous Security and lecturing Internet Security and Computer Networks at Khalifa University in UAE as an Assistant Professor.

Taeshik Shon is a senior engineer in the Convergence Solution Team, DMC R&D Center of Samsung Electronics Co., Ltd. He received his PhD degree in Information Security from Korea University, Seoul, Korea, 2005 and his MS and BS degrees in computer engineering from Ajou University, Suwon, Korea, 2000 and 2002, respectively. While he was working toward his PhD degree, he was awarded a KOSEF scholarship to be a research scholar in the Digital Technology Center, University of Minnesota, Minneapolis, U.S.A., from February 2004 to February 2005. He was awarded the Gold Prize for the Sixth Information Security Best Paper Award from the Korea Information Security Agency in 2003, the Honorable Prize for the 24th Student Best Paper Award from Microsoft-KISS, 2005, the Bronze Prize for the Samsung Best Paper Award, 2006, and the Second Level of TRIZ Specialist certification in compliance with the International TRIZ Association requirements, 2008. He is also serving as an editorial staff and review committee of the Journal of The Korea Institute of Information Security and Cryptology, IAENG International Journal of Computer Science, and other journals. His research interests include Mobile/Wireless Network Security, WPAN/WSN Network Security, network intrusion detection systems, and machine learning.

Jonghyuk Park received his PhD degree from the Graduate School of Information Security from Korea University, Korea. From December, 2002 to July, 2007, Dr. Park had been a research scientist in R&D Institute, Hanwha S&C Co., Ltd., Korea. From September, 2007 to August, 2009, he had been a professor in the Department of Computer Science and Engineering, Kyungnam University, Korea. He is now a professor at the Department of Computer Science and Engineering, Seoul National University of Technology, Korea. Dr. Park has published about 100 research papers in international journals and conferences. He has been serving as chairs, program committee, or organizing committee chair for many international conferences and workshops.

Kwangjo Kim received the BS and MS degrees in Electronic Engineering from Yonsei University, Korea, and the PhD in the Div. of Electrical and Computer Engineering from Yokohama National University, Japan. Currently he is a Professor at the School of Computer Science in KAIST, Korea. He is also the president of the Korean institute of Information Security and Cryptography.

DOI: 10.1002/dac